# Good quantum-convolutional error-correction codes and their decoding algorithm exist

H. F. Chau*
*Department of Physics, University of Hong Kong, Pokfulam Road, Hong Kong*
(Received 8 January 1999)

A quantum-convolutional code was introduced recently as an alternative way to protect vital quantum information. To complete the analysis of the quantum-convolutional code, I report a way to decode certain quantum-convolutional codes based on the classical Viterbi decoding algorithm. This decoding algorithm is optimal for a memoryless channel. I also report three simple criteria to test if decoding errors in a quantum-convolutional code will terminate after a finite number of decoding steps whenever the Hilbert space dimension of each quantum register is a prime power. Finally, I show that certain quantum-convolutional codes are in fact stabilizer codes. And hence, these quantum stabilizer convolutional codes have fault-tolerant implementations. [S1050-2947(99)08909-X]

PACS number(s): 03.67.Dd, 89.70.+c, 89.80.+h

## I. INTRODUCTION

Quantum error-correcting codes (QECCs) and their fault-tolerant implementations are effective ways to protect and to manipulate quantum information in the presence of noise. A QECC works by adding suitable redundancy in the form of entanglement to the original quantum state in such a way that one can reconstruct the original state after decoherence. Since the discovery of QECCs by Shor [1], researchers have discovered many ways to construct QECCs. (See, for example, Refs. [2–12].) These QECCs share a common characteristic, namely, one first divides the original quantum registers into separate blocks of a fixed finite length. One then applies the same encoding method to quantum registers in each block. Hence, this kind of code is called quantum block codes (QBCs). A QECC can be decoded by first measuring the error syndromes of the encoded quantum state and then by applying a necessary unitary transformation to the corresponding erroneous quantum registers [1,6]. For QBCs, this can be carried out in a block-by-block basis. Since there is only a finite number of error syndromes and hence also a finite number of recovery operations in each block, decoding a QBC requires only a finite amount of work per (decoded) quantum register.

Recently, Chau constructed another class of codes, known as quantum-convolutional codes (QCCs), whose encoding scheme for each block depends on the states of several other blocks [13]. For example, he showed that the QCC

$$
\begin{aligned}
|k_1,k_2,\ldots\rangle \mapsto & |\mathbf{k}_{\mathrm{L}}\rangle \\
\equiv & \bigotimes_{i=1}^{+\infty}\Bigg[\sum_{p_1,q_1,\ldots}\frac{1}{N}\omega_N^{(k_i+k_{i-2})p_i+(k_i+k_{i-1}+k_{i-2})q_i} \\
& \times|p_i+p_{i-1},p_i+p_{i-1}+q_{i-1},q_i \\
& +q_{i-1},q_i+q_{i-1}+p_i\rangle\Bigg], \qquad (1)
\end{aligned}
$$

where $k_i\in\mathbb{Z}_N$ for all $i>0$, $k_j=0$ for all $j\leqslant 0$, $\omega_N$ is a primitive $N$th root of unity, the sum is from 0 to $N-1$, and all additions in the state ket are modulo $N$, is capable of correcting one error out of every eight consecutive quantum registers.

While QCCs are of interest of their own right, it is not clear how to decode them. This is because the length of the original quantum state and hence the number of correctable errors by the code may both be infinite. Furthermore, decoding errors may propagate from one block to another due to their convolutional nature. Besides, it is not obvious how to manipulate a QCC in a way that is tolerant of faults.

In this paper, I address the questions of decoding and manipulating a QCC in a way that is tolerant of fault as well as a condition for the existence of a QCC whose decoding error does not propagate indefinitely. My key observation is that many classical convolutional code as well as quantum block code concepts can be extended to the quantum case when one performs the relevant operations carefully. I first show that the well-known Viterbi decoding algorithm (VDA) [14,15] for classical convolutional codes can be generalized to QCCs. Then, I show that the quantum version of the Viterbi decoding algorithm (QVA) is equivalent to the maximum likelihood decoding. And hence the QVA is optimal in a memoryless channel. After that, I investigate the decoding error propagation in QCCs. In particular, I prove three equivalent criteria for QCCs to have finite decoding error propagation whenever the Hilbert space dimension of each quantum register is a prime power. And finally I address the question of fault-tolerant manipulation of QCCs by showing that the well-known fault-tolerant stabilizer code theory can be generalized to QCCs.

## II. CLASSICAL AND QUANTUM VITERBI DECODING ALGORITHMS

Before I go on, let me stress that in my subsequent discussion, I shall restrict myself to considering those classical or quantum-convolutional codes whose encoding can be implemented by a $k$-input $n$-output (and hence also $n-k$ preset registers in the quantum case) $m$-memory (that is, the encoding scheme depends on the state of the previous $m$ blocks) quantum sequential circuit. (Compare with the defi-

*Electronic address: hfchau@hkusua.hku.hk

 

nition of classical convolutional codes in Ref. [16].) Actually, all useful convolutional codes belong to this category.

Now, let me begin by briefly reviewing VDA for binary signals [14–17]. The algorithm starts by computing the Hamming distances between the first $dn$ bits of the signal in the encoded sequence with the first $dn$ bits of the $2^{dk}$ possible code branches where $d=\lfloor km/(n-k)+1 \rfloor$. One then keeps only those $2^{(d-1)k}$ code branches with small Hamming distances. (In case of a tie in Hamming distances, one keeps the corresponding code branches arbitrarily.) Now, one computes the Hamming distances between the first $(d+1)n$ bits of signal with the first $(d+1)n$ bits of all possible code branches that are consistent with a previously kept code branch. This process is repeated until either the signal ends for signals of finite length or the process is repeated definitely for an infinitely long signal. The final surviving code branch is the decoded signal. In essence, the VDA tries to find a codeword with the smallest Hamming distance from the signal [15–21].

Clearly, the VDA in the above form cannot be applied directly to QCCs as it requires a complete knowledge of the encoded signal. However, if one examines the algorithm carefully, it becomes clear that what are really required are the *error syndromes* of the encoded signal for different possible code branches. Consequently, the VDA can be applied to quantum signals. Suppose $\mathcal{R}$ spans of the set of all error recovery (unitary) operators for every $n$ consecutive quantum registers. [For a general error correcting code, $\mathcal{R}$ can be chosen to be in the form $\otimes_{i=1}^{k}(f_i \circ s_i)$ where $f_i$ is either an identity or a controlled phase-shift operator on a single quantum register, and $s_i$ is either an identity or a spin permutation operator on a single quantum register. Consequently, there are $(N\,N!)^k$ elements in $\mathcal{R}$.] Recall that error syndromes can be regarded as operators whose actions have no effect on an error-free encoded quantum state. (For instance, the action of error syndrome on a stabilizer code simply permutes the stabilizer.) Thus, by measuring the eigenvalue of an error syndrome, one gains some information on the location and kind of error that occurred in a quantum code [3,6]. Since the set of all error syndromes is closed under composition and the QCC is of finite memory, there is only a finite number of independent error syndrome operators that acts only on the first $d$ blocks of encoded quantum registers. So, for each $R \in \mathcal{R}^{\otimes d}$, I use a finite (and fixed) number of ancillary quantum registers to measure the error syndromes of the first $dn$ quantum registers after subjecting them to the unitary operation $R$. In this way, I can locate the erroneous registers for each $R \in \mathcal{R}^{\otimes d}$.

Once the erroneous quantum registers are located, how are we going to correct the quantum errors? To answer this question, I have to introduce the following definition first.

*Definition 1.* Let $|\Psi_1\rangle$ and $|\Psi_2\rangle$ be two quantum signals of possibly infinite lengths. If it is not possible to find a unitary operator involving a finite number of quantum registers which maps $|\Psi_1\rangle$ to $|\Psi_2\rangle$, then I say that the *quantum Hamming distance* between $|\Psi_1\rangle$ and $|\Psi_2\rangle$ is infinite. Otherwise, I define the *quantum Hamming distance* between these two quantum signals as the minimum number of quantum registers involved in unitarily transforming from one state to the other.

Similarly, I define the minimum quantum Hamming distance between a quantum signal $|\Psi\rangle$ and the set of all possible code words of a QECC $C$ to be infinite if the quantum Hamming distances between $|\Psi\rangle$ and all code words of $C$ are infinite. Otherwise, I define the quantum Hamming distance between $|\Psi\rangle$ and $C$ to be the minimum possible quantum Hamming distance between $|\Psi\rangle$ and the code words in $C$. And for simplicity, I shall simply call the minimum quantum Hamming distance between $|\Psi\rangle$ and $C$ the *quantum Hamming distance* of $|\Psi\rangle$.

I also define the *recovery cost* of bringing the first $dn$ quantum signals with respected to $R \in \mathcal{R}^{\otimes d}$ to be the quantum Hamming distance of the quantum signal plus the (minimum) number of registers affected by $R$.

Readers can easily check that quantum Hamming distance between two quantum signals is a metric for the set of all quantum signals. Moreover, in a loose sense, the recovery cost measures how close and how much work is required to bring a quantum signal to a quantum codeword.

With the above definition, I am ready to report QVA decoding: By carefully measuring the error syndromes using ancillary quantum registers, I compute the recovery cost for the first $dn$ quantum registers for each $R \in \mathcal{R}^{\otimes d}$. I keep $|\mathcal{R}|^{d-1}$ error recovery operators with small recovery costs out of the $|\mathcal{R}|^d$ possible ones (where $|\mathcal{R}|$ denotes the number of elements in the set $\mathcal{R}$). Then, I go on to compute the recovery cost for the first $(d+1)n$ quantum signals with respect to the set of all possible recovery operators in $\mathcal{R}^{\otimes(d+1)}$ that are consistent with a previously kept recovery operator in $\mathcal{R}^{\otimes d}$. For a quantum signal of finite length, I repeat this process until the quantum signal terminates. Then, I regard the error of the signal to be caused by the one that produces the minimum possible recovery cost among those $|\mathcal{R}|^d$ ones I kept at the end. And I correct the quantum signal accordingly. For quantum signals of infinite length, I have to repeat the recovery cost selection process infinitely many times in order to find the minimum recovery cost path. In practice, we usually run the QVA over a large but finite number of quantum blocks and decode the signal in each block separately. The length of such a quantum block is usually adaptive; that is to say, it is chosen in such a way that the recovery cost paths retained do not differ very much from each other. In this way, the effect due to the choice of the length of the block is minimized. In summary, regardless of the length of the quantum signal, the QVA uses a finite number of operations on average to recover it.

Finally, I need to convert the recovered encoded signal to its unencoded form. Since I may have an infinitely long signal, the usual trick of running the reversible encoding quantum circuit backward does not work. Hence, I have no choice but to decode the signal starting from the first encoded block. Remember that by including the preset quantum registers, the encoding process can be represented by a unitary transformation. Let $C$ be a $k$-input $n$-output $m$-memory QCC and $|\Psi_1\rangle$ be a quantum signal (with preset quantum registers added). I denote the encoding process for this quantum code $C$ and quantum signal $|\Psi\rangle$ by $U_1$. Moreover, I denote the encoding process by the same code $C$ on the quantum signal $|\Psi_i\rangle$ by $U_i$ where $|\Psi_i\rangle$ equals $|\Psi_1\rangle$ except that the first $k(i-1)$ quantum registers are set to zero. Using this nota-

tion, I write the encoding process $U_1$ as $(U_1 \circ U_2^{-1}) \circ (U_2 \circ U_3^{-1}) \circ \cdots$. Consequently, the decoding process is given by

$$\begin{aligned} U_1^{-1} &= \cdots \circ (U_3 \circ U_4^{-1})^{-1} \circ (U_2 \circ U_3^{-1})^{-1} \circ (U_1 \circ U_2^{-1})^{-1} \\ &= \cdots \circ (U_4 \circ U_3^{-1}) \circ (U_3 \circ U_2^{-1}) \circ (U_2 \circ U_1^{-1}). \end{aligned} \tag{2}$$

Note that for the $m$-memory code $C$, $U_{i+1} \circ U_i^{-1}$ is a unitary operator acting only on the $i$th, $(i+1)$th, up to $(i+m)$th encoded blocks for all $i$. Moreover, it is easy to check that the action of $U_2 \circ U_1^{-1}$ is to extract the state of the first unencoded block of quantum registers out of the encoded state. After that, the action of $U_3 \circ U_2^{-1}$ is to extract the state of the second unencoded block out of the encoded state and so on. Thus, Eq. (2) gives us a way to decode the QCC $C$.

*Example 1.* It is easy to compute $U_{i+1} \circ U_i^{-1}$ in practice. For example, to decode the QCC in Eq. (1) in Eq. (3a), I subtract the second quantum register by the first, the fourth by the first and the third, the fifth by the first, the sixth by the first and the fourth, the seventh by the third, and finally the eighth by the third. The resultant quantum state is given by Eq. (3b). Then, I discretely inverse Fourier transform the first quantum register. The resultant state is given by Eq. (3c) after summing over the dummy index $p_1$. Finally, it is straightforward to unitarily convert the state in Eq. (3c) to the state in Eq. (3d) by multiplying a phase proportional to the product of the first quantum register by the sum of the third and the 12th registers, and then followed by discrete Fourier transforming the third quantum register. The result of all these steps above gives $U_2 \circ U_1^{-1}$:

$$\begin{aligned} &\sum_{p_1,q_1,\ldots} \frac{1}{N} \omega_N^{k_1(p_1+q_1)} |p_1,p_1,q_1,p_1+q_1\rangle \\ &\quad \otimes \frac{1}{N} \omega_N^{k_2 p_2 + (k_2+k_1) q_2} |p_1+p_2, p_1+p_2+q_1, q_2+q_1, p_2+q_2+q_1\rangle \\ &\quad \otimes \frac{1}{N} \omega_N^{(k_3+k_1)p_3 + (k_3+k_2+k_1)q_3} |p_3+p_2, p_3+p_2+q_2, q_3+q_2, p_3+q_3+q_2\rangle \otimes \cdots \end{aligned} \tag{3a}$$

$$\begin{aligned} &\mapsto \sum_{p_1,q_1,\ldots} \frac{1}{N} \omega_N^{k_1(p_1+q_1)} |p_1,0,q_1,0\rangle \\ &\quad \otimes \frac{1}{N} \omega_N^{k_2 p_2 + (k_2+k_1) q_2} |p_2,p_2,q_2,p_2+q_2\rangle \\ &\quad \otimes \frac{1}{N} \omega_N^{(k_3+k_1)p_3 + (k_3+k_2+k_1)q_3} |p_3+p_2, p_3+p_2+q_3, q_3+q_2, p_3+q_3+q_2\rangle \otimes \cdots \end{aligned} \tag{3b}$$

$$\begin{aligned} &\mapsto \sum_{\lambda,p_1,q_1,\ldots} \frac{1}{N^{3/2}} \omega_N^{k_1(p_1+q_1)-p_1\lambda} |\lambda,0,q_1,0\rangle \\ &\quad \otimes \frac{1}{N} \omega_N^{k_2 p_2 + (k_2+k_1) q_2} |p_2,p_2,q_2,p_2+q_2\rangle \\ &\quad \otimes \frac{1}{N} \omega_N^{(k_3+k_1)p_3 + (k_3+k_2+k_1)q_3} |p_3+p_2, p_3+p_2+q_3, q_3+q_2, p_3+q_3+q_2\rangle \otimes \cdots \\ &= \sum_{q_1,p_2,q_2,\ldots} \frac{1}{\sqrt{N}} \omega_N^{k_1 q_1} |k_1,0,q_1,0\rangle \otimes \frac{1}{N} \omega_N^{k_2 p_2 + (k_2+k_1) q_2} |p_2,p_2,q_2,p_2+q_2\rangle \\ &\quad \otimes \frac{1}{N} \omega_N^{(k_3+k_1)p_3 + (k_3+k_2+k_1)q_3} |p_3+p_2, p_3+p_2+q_3, q_3+q_2, p_3+q_3+q_2\rangle \otimes \cdots \end{aligned} \tag{3c}$$

$$\begin{aligned} &\mapsto \sum_{p_2,q_2,\ldots} |k_1,0,0,0\rangle \otimes \frac{1}{N} \omega_N^{k_2(p_2+q_2)} |p_2,p_2,q_2,p_2+q_2\rangle \\ &\quad \otimes \frac{1}{N} \omega_N^{k_3 p_3 + (k_3+k_2) q_3} |p_3+p_2, p_3+p_2+q_2, q_3+q_2, p_3+q_3+q_2\rangle \otimes \cdots \end{aligned} \tag{3d}$$

In this way, I decode the first quantum register from the code using a finite number of two-body operators. And inductively I can decode the rest of the encoded quantum signal efficiently.

Now, I move on to prove the optimality of the QVA for a memoryless quantum channel, namely, a noisy quantum channel whose error occurs randomly and independently. Similar to the VDA, the QVA tries to search for a solution in the code word space with a minimum recovery cost from the signal. (I choose to minimize the recovery cost instead of simply the quantum Hamming distance because certain fault-tolerant operations $U$ for the QCC may belong to $\otimes_{i=1}^{+\infty}\mathcal{R}$. In this case, the quantum Hamming distances of $|\Psi\rangle$ and $U|\Psi\rangle$ agree although they have different recovery costs.) After transmitting an arbitrary unknown encoded quantum state through a memoryless channel, the effect of decoherence can be regarded as a Markov process. In other words, the probability that the error recovery operator needed to act on the $(t+1)$th block of quantum signals, given all error recovery operators for the first $t$ blocks, depends only on that of the $(t+1)$th block. Since I do not manipulate my encoded quantum signal during its transmission, I can always conceptually regard my error syndrome measurement to be performed immediately after the errors are introduced into the signal. But once I have measured the error syndromes, the location as well as the type of error each quantum register is suffering from becomes classical data. Therefore, the effect of a quantum memoryless channel is the same as that in a classical probabilistic memoryless channel. More precisely, I can always model the chance for a certain quantum error $R\in\mathcal{R}$ to error in a quantum register by a classical probability function. (Compare to the argument used in the proof of the security of quantum key distributions in Ref. [22].) Consequently, the optimality proof of the VDA [15–19] carries over directly to the QVA.

Similar to classical convolutional codes, there are two important probabilities which measure the performance of a QCC. The first one is called the error probability $P_e(E)$, which is defined to be the probability that a wrong decoding path is chosen at any given time step. And the second one is called the qubit error probability $P_b(E)$, which is defined to be the expected number of information qubit decoding errors per decoded information qubit. For $N=2$ in a binary symmetric channel, these two probabilities are given by [17]

$$P_e(E)\lesssim A_d 2^d p^{t/2} \tag{4a}$$

and

$$P_b(E)\lesssim \frac{B_d}{k}2^d p^{d/2}, \tag{4b}$$

where $d$ is the minimum quantum Hamming distance between code words, $A_d$ is the number of mutually orthogonal encoded states of quantum Hamming weight $d$, $B_d$ is the number of nonzero mutually orthogonal information qubits on all weight $d$ paths, $k$ is number of information qubits per block, and $p$ is error probability of the channel.

For a fixed QCC, the distance of the code $d$ is finite and hence both $P_e$ and $P_b$ scale only like a power law of $p$. Nevertheless, there are many $k$-input $n$-output QCCs with different memories $m$. And for many fixed $k$ and $n$, the distance of the code $d$ increases approximately linearly with memory $m$. And such a family of codes may be constructed, for instance, from a corresponding family of classical convolutional codes. One such family of classical codes as briefly discussed in Ref. [17] encodes one classical bit into two. By the construction of Chau in Ref. [13], it can be turned into a family of one-input four-output QCCs. And as the memory $m$ tends to infinity, both $P_e$ and $P_b$ become exponentially small. Thus, we have a family of good QCCs.

## III. NONCATASTROPHIC QUANTUM CODES

The ability to decode a QCC is not sufficient to make QCCs useful. We must also make sure that any decoding error will not propagate infinitely in spite of the convolutional nature of the code. To facilitate discussions, I borrow the following terminology from classical coding theory.

*Definition 2.* A QCC is said to be *catastrophic* if there exists a local decoding error that can propagate infinitely. Otherwise, a QCC is said to be *noncatastrophic*. Clearly, useful QCCs must be noncatastrophic.

In the case of classical convolutional codes and when the number of internal states per register $N$ is a prime power, a convolutional encoder can be mathematically represented by a polynomial of one variable over a finite field. Such a polynomial ring is clearly a Euclidean domain. In particular, two polynomials in a Euclidean domain have a unique greater common divisor (up to multiplication of units). Using this nice property of a Euclidean domain, Massey and Sain [25,26] proved a necessary and sufficient condition for a classical convolutional code to be noncatastrophic. Nonetheless, quantum mechanical operations are intrinsically noncommutative. Thus, the proof of Massey and Sain does not work for QCCs.

Quite surprisingly, a necessary and sufficient condition for a QCC to be noncatastrophic can still be found whenever $N$ is a prime power (and hence $\mathbb{Z}_N$ is a finite field). And I am going to report the criterion after introducing the following rather involved notation.

### A. Notation for encoding and decoding qubits when $N$ is a prime power

In the case $N$ is a prime power, any two-body unitary operation can be generated by the span of the following elementary Pauli group operations [23]: (a) addition $|x\rangle\mapsto|x+a\rangle$ and $|x,y\rangle\mapsto|x+y,y\rangle$ for some $a\in\mathbb{Z}_N$, (b) multiplication $|x\rangle\mapsto|ax\rangle$ for some $a\in\mathbb{Z}_N\backslash\{0\}$, (c) Fourier transform $|x\rangle\mapsto\Sigma_y\omega_N^{xy}|y\rangle$, (d) local phase multiplication $|x\rangle\mapsto\omega_N^{ax}|x\rangle$, and (e) nonlocal phase multiplication $|x,y\rangle\mapsto\omega_N^{xy}|x,y\rangle$.

Let me first consider those QCCs that can be encoded and can be implemented by a $k$-input $n$-output $m$-memory quantum sequential circuit. In this case, I can group the initial unencoded quantum state and preset registers into blocks of length $n$. This state is spanned by $\{\otimes_{i=1}^{+\infty}|x_{i1},x_{i2},\ldots,x_{in}\rangle\}$, where $x_{ij}$ for $i\geqslant 1$ and $1\leqslant j\leqslant n-k$ are the preset registers and $x_{ij}$ for $i\geqslant 1$ and $n-k<j\leqslant n$ are the quantum information registers. With this notation in mind, I define the following operators: (a) state delay operator $D^m:|x_{i,j}\rangle\mapsto|x_{i-m,j}\rangle$, (b) Fourier transform operator $F_p:|x_{ij}\rangle$

$\mapsto\Sigma_{x_{ij}}\omega_N^{x_{ij}y_{i,j-p}}|y_{i,j-p}\rangle$, (c) phase projection operator[1] $P:e^{i\phi}|x_{ij}\rangle\mapsto|x_{ij}\rangle$, (d) local phase multiplication operator $L^m:|x_{ij}\rangle\mapsto\omega_N^{mx_{ij}}|x_{ij}\rangle$, (e) nonlocal phase multiplication operator $M_{mp}:|x_{i,j}\rangle\mapsto\omega_N^{x_{i,j}x_{i-m,j-p}}|x_{i,j}\rangle$, (f) state addition operator $+:(e^{i\phi}|x\rangle,e^{i\phi'}|x'\rangle)\mapsto e^{i(\phi+\phi')}|x+x'\rangle$, and (g) state multiplication operator $a:|x_{ij}\rangle\mapsto|ax_{ij}\rangle$. Using this notation, the five elementary Pauli group operations can be represented by (a) $1+D^mP$, (b) $a$, (c) $F_0$, (d) $L^a$, and (e) $M_{m0}$, respectively. More generally, if I write the initial state ket together with the preset and ancillary quantum registers in a $p\times1$ column vector, then the composition of several Pauli group operations can be represented by a $p\times p$ matrix whose elements belong to the noncommutative ring $\mathbb{Z}_N\langle D,M_{ij},P,L,F_i\rangle$ with $F_i^2=-1$, $P^2=P$, $PD=DP$, $LM_{ij}=M_{ij}L$, $PL=P$, $PM_{ij}=P$, $M_{ij}M_{pq}=M_{pq}M_{ij}$, and $M_{ij}m=m^{-1}M_{i,j}^m$ for $m\in\mathbb{Z}_N\backslash\{0\}$.

Since the Pauli group spans the set of all two-body operators [23], a general quantum encoding circuit $U_{\text{encode}}$ for a $k$-input $n$-output $m$-memory QCC can be written as a finite sum $\Sigma_i(\alpha_i,g_i)$ where $\alpha_i\in\mathbb{C}$ and $g_i\in(\mathbb{Z}_N\langle D,M_{ij},P,L,F_i\rangle)^{n+p,n+p}$ where $p$ is the number of ancillary quantum registers required in the encoding process per block. For instance, the operator $\Sigma_{i=0}^{N-1}(1/N,L^i)$ sends $|0\rangle$ to $|0\rangle$ and all other $|i\rangle$ to 0. Furthermore, the unitary operator sending $|x,y\rangle$ to $|x,x+y\rangle$ can be written as

$$\begin{bmatrix}(1,1) & (0,0)\\ (1,P) & (1,1)\end{bmatrix}. \tag{5}$$

Readers should observe that the phase projection operator $P$ in Eq. (5) is essential. If I replace $P$ by 1 in Eq. (5), then the replaced operator will not be well defined for it would have mapped $e^{i\theta_1+\theta_2}|x,y\rangle$ to $e^{i\theta_1+\theta_2}|x,x+y\rangle$ and $e^{i\theta_1}|x\rangle\otimes e^{i\theta_2}|y\rangle$ to $e^{i(2\theta_1+\theta_2)}|x,x+y\rangle$. In addition, the operator expressed in Eq. (5) is unitary in spite of its apparent nonskew symmetric form.

Let me denote the set of all finite sums in the form $\Sigma_i(\alpha_i,g_i)$ by $K$. Then, if I forget about the initial preset and ancillary quantum registers and simply represent the initial unencoded quantum information as a $k\times1$ column vector, then I can simply write a $k$-input $n$-output $m$-memory quantum encoding circuit as a $n\times k$ matrix in $K^{n,k}$. The decoding circuit for this QCC is equal to a $(n+p)\times(n+p)$ matrix $U_{\text{encode}}^{-1}$. Nevertheless, $U_{\text{encode}}^{-1}\notin K^{n+p,n+p}$ in general. Similar to the encoding circuit, if I forget about the initial preset and ancillary quantum registers used in the decoding circuit, then I can present the decoding circuit by a $k\times n$ matrix.

*Example 2.* Using the above notation, the encoding and decoding algorithms for the classical noncatastrophic convolutional code (written in a quantum state ket form) $|k_1,k_2,\ldots\rangle\mapsto\otimes_{i=1}^{+\infty}|k_i+k_{i-2},k_i+k_{i-1}+k_{i-2}\rangle$ can be written as

$$\begin{bmatrix}(1,P[1+D^2])\\ (1,D+P[1+D^2])\end{bmatrix} \tag{6}$$

and

$$[(1,-D^{-1})\quad(1,D^{-1})], \tag{7}$$

respectively.

Similarly, the encoding and decoding algorithms for the classical catastrophic convolutional code $|k_1,k_2,\ldots\rangle\mapsto\otimes_{i=0}^{+\infty}|k_i+k_{i-1},k_i+k_{i-2}\rangle$ can be written as

$$\begin{bmatrix}(1,P[1+D])\\ (1,1+PD^2)\end{bmatrix} \tag{8}$$

and

$$[(1,[1+PD]^{-1}D)\quad(1,[1+PD]^{-1})], \tag{9}$$

respectively.

*Example 3.* One possible way to encode the quantum state $|k_1,k_2,\ldots\rangle$ as a QCC given by Eq. (1) is as follows: First, I prepare a number of preset quantum registers and write the initial state as $\otimes_{i=1}^{+\infty}|k_i,0,0,0\rangle$. Then, I transform this state to $\otimes_{i=1}^{+\infty}|k_i,k_{i-1},k_{k-2},0\rangle$ by the unitary operator

$$\begin{bmatrix}(1,1) & (0,0) & (0,0) & (0,0)\\ (1,PD) & (0,0) & (0,0) & (0,0)\\ (1,PD^2) & (0,0) & (1,1) & (0,0)\\ (0,0) & (0,0) & (0,0) & (1,1)\end{bmatrix}. \tag{10}$$

Then, I transform the state to $\otimes_{i=1}^{+\infty}|k_i+k_{i-2},k_i+k_{i-1}+k_{i-2},0,0\rangle$ by

$$\begin{bmatrix}(1,1) & (0,0) & (1,PD) & (0,0)\\ (1,PD) & (1,1) & (1,PD^2) & (0,0)\\ (0,0) & (1,-PD) & (1,1) & (0,0)\\ (0,0) & (0,0) & (0,0) & (1,1)\end{bmatrix}. \tag{11}$$

Next, I unitarily transform the state to $\otimes_{i=1}^{+\infty}\omega_N^{p_i(k_i+k_{i-2})+q_i(k_i+k_{i-1}+k_{i-2})}|p_i,q_i,0,0\rangle$ by

$$\begin{bmatrix}(1,F_0) & (0,0) & (0,0) & (0,0)\\ (0,0) & (1,F_1) & (0,0) & (0,0)\\ (0,0) & (0,0) & (1,1) & (0,0)\\ (0,0) & (0,0) & (0,0) & (1,1)\end{bmatrix}. \tag{12}$$

Finally, I bring the state to Eq. (1) by the unitary transformation

$$\begin{bmatrix}(1,P[1+D]) & (0,0) & (1,1) & (0,0)\\ (1,P[1+D]) & (1,PD) & (1,P) & (1,1)\\ (0,0) & (1,1) & (0,0) & (1,P)\\ (1,1) & (1,P) & (0,0) & (1,P)\end{bmatrix}. \tag{13}$$

Thus, the unitary encoding transformation $U_{\text{encode}}$ for the QCC in Eq. (1) simply equals the product of the matrices in Eqs. (10)–(13). Moreover, if we forget about the initial pre-

[1]I shall explain why I introduce such a noninvertible projection operator in the next paragraph.

set registers, then the encoding operation is simply given by the first column of the matrix $U_{\text{encode}}$, which is given by

$$\begin{bmatrix} (1,P[1+D]F_0[1+PD^2]) \\ (1,P[1+D]F_0[1+PD^2]+PDF_1[1+D+D^2]) \\ (1,[1+PD]F_1P[1+D+D^2]) \\ (1,F_0[1+PD^2]+P[1+D]F_1P[1+D+D^2]) \end{bmatrix}. \tag{14}$$

Similarly, the decoding operation is equal to the first row of the matrix $U_{\text{encode}}^{-1}$, namely,

$$\begin{bmatrix} (1,P[DF_1^{-1}-1]) \\ (1,-PDF_1) \\ (1,[1+PD]F_0P-PDF_1^{-1}-P[1+D]) \\ (1,[1+PD]F_0^{-1}+P[1+D]) \end{bmatrix}^{\mathrm{T}}$$

$$=\begin{bmatrix} (1,P[-DF_1(-1)-1]) \\ (1,-PDF_1) \\ (1,[1+PD]F_0P+PDF_1(-1)-P[1+D]) \\ (1,-[1+PD]F_0(-1)+P[1+D]) \end{bmatrix}^{\mathrm{T}}. \tag{15}$$

### B. Criterion for noncatastrophic quantum code when $N$ is a prime power

Now, let me report a useful lemma before proving a necessary and sufficient condition for noncatastrophic QCCs.

*Lemma 1.* Suppose $N$ is a prime power. And let $M \in K^{p,p}$ be a valid unitary operator acting on a possible infinitely long quantum signal. Then, $M$ can be decomposed into a product of finite product $\Pi_{i=1}^{q} M_i$. Moreover, the $p^2$ elements in each matrix $M_i$ commute with each other for all $i$.

*Proof.* Since $N$ is a prime power and hence $\mathbb{Z}_N$ is a field, I can borrow the idea in Ref. [24] to decompose the matrix $M$ as a product of finitely many matrices. Observe that I can always find an invertible $p\times p$ matrix $N_1$ such that the element located in the $p$th row and $(p-1)$th column in $N_1^{-1}M$ equals (0,0). Besides, I can choose $N_1$ in such a way that the elements in the $i$th row and $j$th column satisfy the condition

$$(N_1)_{ij}=\begin{cases} (1,1) & \text{if } i=j \text{ and } i\leqslant p-2 \\ (0,0) & \text{if } i\neq j \text{ and } (i,j)\neq(p,p-1). \end{cases} \tag{16}$$

Similarly, I can find a $p\times p$ matrix $N_2$ such that $(N_2^{-1}N_1^{-1}M)_{ij}=(0,0)$ whenever $(i,j)=(p,p-1)$ and $(p,p-2)$. Moreover,

$$(N_2)_{ij}=\begin{cases} (1,1) & \text{if } i=j \text{ and } i\neq p-2 \text{ or } p \\ (0,0) & \text{if } i\neq j \text{ and } (i,j)\neq(p,p-2). \end{cases} \tag{17}$$

Inductively, I can find $N_i$ such that $M' = N_{p(p-1)/2}^{-1}N_{[p(p-1)/2]-1}^{-1}\cdots N_1^{-1}M$ is an upper triangular matrix. Besides, at most three elements in $N_i-I_p$ are nonzero where $I_p$ denotes the identity operator. Similarly, I can transform the matrix $M'$ into a diagonal one by means of $p(p-1)/2$ matrices in a similar form as $N_i$. Thus,

$$M=N_1N_2\cdots N_{p(p-1)/2}M''N_{[p(p-1)/2]+1}N_{[p(p-1)/2]+1}N_1N_2 \times\cdots N_{p(p-1)/2}M''N_{[p(p-1)/2]+1}N_{[p(p-)/2+2}\cdots N_{p(p-1)},$$

where $M''$ is a diagonal matrix and all $N_i-I_p$ can be brought into the form in Eq. (16) by relabeling some columns and rows plus possibly a transposition.

It is obvious that $M''$ is equal to a product of $p$ diagonal matrices, each of which has at most one diagonal element different from (1,1). Besides, elements in each of the $p$ matrices commute with each other. Thus, to complete the proof, it remains to show that $N_i$ can be decomposed into a finite product of matrices whose elements commute. In fact, it suffices for me to decompose it for the $2\times 2$ matrix

$$\tilde{M}=\begin{bmatrix} A & B \\ (0,0) & C \end{bmatrix}.$$

The decomposition for the matrix $N_i$ is similar. Since $\tilde{M}$ is a well-defined operator, either $A$ or $B$ (but not both) must be in the form $PX$ for some $X\in K$. In the first case, $A=PX$ and it is easy to check that

$$\tilde{M}=\begin{bmatrix} (1,0) & (0,0) \\ (0,0) & C \end{bmatrix}\begin{bmatrix} (1,P) & B \\ (0,0) & (1,0) \end{bmatrix}\begin{bmatrix} X & (0,0) \\ (0,0) & (1,1) \end{bmatrix}. \tag{18a}$$

And in the second case, $B=PY$ and

$$\tilde{M}=\begin{bmatrix} (1,0) & (0,0) \\ (0,0) & C \end{bmatrix}\begin{bmatrix} (1,P) & Y \\ (0,0) & (1,0) \end{bmatrix}\begin{bmatrix} X & (0,0) \\ (0,0) & (1,1) \end{bmatrix}. \tag{18b}$$

Since elements in each of the matrices on the right hand sides of Eqs. (18a) and (18b) commute, so the lemma is proved.

After going through the above preparatory discussions and examples, I am ready to state a necessary and sufficient condition for noncatastrophic QCCs. In fact, theorem 1 below generalizes a necessary and sufficient condition for classical noncatastrophic codes [25,26].

*Theorem 1.* The following statements concerning a $k$-input $n$-output $m$-memory QCC are equivalent when $N$ is a prime power.

(a) The QCC is noncatastrophic.

(b) There exits a quantum encoding circuit (which includes the preset and ancillary quantum registers) $g \in K^{n+p,n+p}$ for the QCC such that its left inverse $g^{-1}$ exists. Moreover, elements in the matrix $g^{-1}$ can be expressed as a finite sum $\Sigma_i(\alpha_i,g_i')$ with $g_i'\in\mathbb{Z}_N\langle D,D^{-1},M_{pq},P,L,F_p\rangle$ for all $i$.

(c) There exists a quantum encoding circuit that can be decomposed into the finite product $g=\Pi_i g_i$ in such a way that for each $i$, (1) $g_i\in K^{n+p,n+p}$, (2) elements of matrix $g_i$ belong to a commutative polynomial ring, and (3) the inverse $(\det g_i)^{-1}$ exists and can be expressed as a finite sum $\Sigma_i(\alpha_j,g_j')$ with $g_j'\in\mathbb{Z}_N\langle D,D^{-1},M_{pq},P,L,F_p\rangle$.

(d) The quantum encoding circuit (which excludes the preset registers) $h\in K^{n,k}$ can be expressed as a finite product of matrices $\Pi_i h_i$ in such a way that for each $i$, (1) $h_i$

$\in K^{a_i,b_i}$, (2) elements of matrix $h_i$ belong to a commutative polynomial ring, and (3) the greatest common divisor $t_i$ of the determinant of all the $\binom{a_i}{b_i}$ submatrices of $h_i$ is invertible and $t_i^{-1}=\Sigma_j(\alpha_j,t'_{ij})$ with $t'_{ij}\in\mathbb{Z}_N\langle D,D^{-1},M_{pq},P,L,F_p\rangle$.

*Proof.* By suitably adding ancillary quantum registers as well as enlarging the encoding matrix to include those ancillary registers, it is easy to see that (d) $\Rightarrow$ (c) $\Rightarrow$ (b). Now, I move on to show that (b) $\Rightarrow$ (a) observes that there is a decoding circuit that can be represented as a $(n+p)\times(n+p)$ matrix whose elements $\Sigma_i(\alpha_i,h_i)$ with $h_i\in\mathbb{Z}_N\langle D,D^{-1},M_{ij},P,L,F_i\rangle$. In other words, decoding each quantum register in the code requires only information from a finite number of encoded quantum registers. Thus, if there is only a finite number of encoded quantum registers in error, then the decoding errors will only be localized in a finite number of quantum registers. Hence, the code is noncatastrophic.

To complete the proof, it remains for me to show that (a) $\Rightarrow$ (d). Recall that if $U_{\text{encode}}\in K^{n,n}$ is the encoding circuit, then the decoding circuit equals $U_{\text{encode}}^{-1}$. So, if statement (d) is false, then I can extend the $k\times n$ decoding circuit into an $n\times n$ one. And since $N$ is a prime power, so by lemma 1, I can conclude that elements in the $k\times n$ decoding circuit are in the form $\Sigma_i(\alpha_i,h_i)$ where $h_i$ belongs to the formal power series noncommutative ring $\mathbb{Z}_N\langle\langle D,M_{ij},P,L,F_i\rangle\rangle$ but not every element in the decoding circuit belongs to $\mathbb{Z}_N\langle D,M_{ij},P,L,F_i\rangle$. Consequently, there exists an encoded quantum register whose state affects the states of infinitely many decoded quantum registers. Thus, the QCC is catastrophic and this completes the proof. ■

Now, it is clear from the proof of theorem 1 that if I first let the encoded quantum to go through the QVA and then I apply the unitary transformation $g^{-1}$ (which is the left inverse of $g$) to it, I can recover the original unencoded quantum information. In addition, it is also clear that any QCC that cannot be expressed as a $k$-input $n$-output $m$-memory sequential quantum circuit must be catastrophic. Moreover, the conclusion in theorem 1 remains valid if I extend the meaning of the $m$-memory QCC to include those QCCs whose encoding scheme depends on the state of a finite number of previous or future blocks.

One possible way to construct a QCC is to start with a classical convolutional code $C$ [13]. Chau showed that one can first encode a quantum signal using the classical code $C$; then one takes the local Fourier transform on each encoded quantum register. And finally one encodes the resultant state ket by the code $C$ again, and one gets a QCC [13]. Here, I show that the QCC generated this way inherits the error propagation behavior from its parent classical code.

*Corollary 1.* Suppose $C$ is a $k$-input $n$-output classical convolutional code and $Q$ be the corresponding $k^2$-input $n^2$-output QCC obtained using the above method. Then $C$ is catastrophic if and only if $Q$ is catastrophic.

*Proof.* I write the quantum encoding scheme as a product of three matrices $g_1g_2g_3$ where $g_1$ and $g_3$ involve the symbols $D$ and $DP$ and $g_2$ involves the symbol $F_i$. That is to say, $g_1$ and $g_3$ represent the initial and final encoding by the classical code $C$ and $g_2$ represents the local Fourier transform. Suppose $C$ is noncatastrophic; then clearly I can arrange $g_1$, $g_2$, and $g_3$ to satisfy statement (d) in theorem 1 [25,26]. Hence, $Q$ is noncatastrophic. Conversely, if $C$ is a catastrophic code, then from the construction of $Q$, it is clear that one can always find a finite number of spin-flip errors for $Q$ such that the decoding errors propagate infinitely. Hence $Q$ is catastrophic. ■

Corollary 1 implies that the QCC given in Eq. (1) is noncatastrophic.

## IV. FAULT-TOLERANT COMPUTATION USING QUANTUM-CONVOLUTIONAL CODES

The ability to decode a noncatastrophic QCC is still not enough to make them truly useful. We have to impose the requirement that the QCC must have a fault-tolerant implementation so that quantum information processing can take place in the encoded form. In QBCs, we know that all stabilizer (block) codes have a fault-tolerant implementation [6,27–33] under a suitable wiring of quantum gates. And now I am going to generalize the theory of the stabilizer code and its fault-tolerant implementations to the world of QCCs.

For stabilizer codes, I restrict myself to considering the case when $N=2$. Recall that in the case of the QBC and when $N=2$, if we denote the coding space of an $n$ qubit code by $T$, then the stabilizer of this code $S$ is some Abelian subgroup of the group $\mathcal{R}^{\otimes n}$ whose elements fixes $T$ [4–6,31,32]. Besides, $S$ can be generated by a finite number of operations $g_i\in\mathcal{R}^{\otimes n}$, known as the generators of $S$. Finally, the encoded spin-flip and phase-change operations are specified in $\mathcal{R}^{\otimes n}$. These operations commute with the stabilizer $S$. More precisely, the code word for a $k$-input $n$-output stabilizer QBC can be written (up to an overall normalization constant) as

$$|x_1,x_2,\ldots,x_k\rangle\mapsto\sum_{q_i=0}^{1}\left[\prod_j\overline{\sigma}_{x,j}\left(\prod_i g_i^{q_i}|0,0,\ldots,0\rangle\right)\right], \tag{19}$$

where $g_i$ and $\overline{\sigma}_{x,j}$ are the generators of the stabilizer and the encoded spin-flip operation for $|x_j\rangle$, respectively [6]. In addition, the encoded phase-shift operators $\overline{\sigma}_{z,j}$ exist in $\mathcal{R}^{\otimes n}$ for all $j$.

Generalizing the stabilizer (block) code formalism to the QCC world is easy. One only needs to be more careful in dealing with the infinite number of qubits and hence the infinite number of generators for the stabilizer. First, one replaces $\mathcal{R}^{\otimes n}$ by $\Pi_{i=1}^{+\infty}\mathcal{R}$. Clearly, $\Pi_{i=1}^{+\infty}\mathcal{R}$ and hence the stabilizer $S$ have a countable number of generators. Thus, Eq. (19) holds for QCCs as $k\to\infty$. Besides, for an $m$-memory QCC, the encoded spin-flip operators $\overline{\sigma}_{x,j}$ as well as the encoded phase-shift operators $\overline{\sigma}_{z,j}$ act on no more than $O(n(m+1))$ qubits. In this way, the fault-tolerant error syndrome measurement procedure in stabilizer block code [6,30] directly applies to the convolutional code. Finally, one concatenates the QCC with another stabilizer QBC to $L$ levels. Then, by correcting the errors in all levels concurrently, one achieves an error reduction from $O(\epsilon)$ to $O(\epsilon^L)$. Hence, Gottesman's [32] proof that all stabilizer codes have fault-tolerant implementation directly carries over to the QCC world. (See also Ref. [33] for related results.) Thus, noncatastrophic stabilizer QCCs are good codes. Recently, Gottesman extended his theory to cover a large number of $N$ary

fault-tolerant quantum codes using the Pauli group [23]. A direct consequence of his result is that we can easily construct many $N$ary fault-tolerant QCCs.

Finally, I go on the show that the QCC in Eq. (1) is a stabilizer code. In fact, I prove something more general.

*Theorem 2.* Let $C$ be a classical convolutional code. And let $Q$ be the corresponding QCC as described in corollary 1. Then, $Q$ is a stabilizer code.

*Proof.* The proof follows directly from the three lemmas below. ■

*Lemma 2.* All classical binary (block or convolutional) codes are stabilizer codes.

*Proof.* Without loss of generality, I consider an $m$-memory classical convolutional code. Then, the encoded spin-flip operator $\overline{\sigma}_{x,j}$ is nothing but a finite number of $\sigma_x$ acting on the encoded qubits. Since the code is classical, the encoded state for each $|x_1,x_2,\ldots\rangle$ can simply be represented by a single state ket without any dummy summation index. More precisely, elements of the stabilizer are those that commute with the encoded spin-flip operators and at the same time can be expressed in the form $A_1A_2A_3\cdots$ where $A_i$ acts on the $i$th encoded qubit and $A_i\in\{\mathbb{1},\sigma_z\}$. Clearly, this kind of element forms an Abelian subgroup of $\Pi_{i=1}^{+\infty}\mathcal{R}$ and has a countable number of generators. Hence, the lemma is proved. ■

*Example 4.* The stabilizer associated with the classical block code $|k\rangle\mapsto|kkk\rangle$ is generated by $\sigma_y\sigma_y\mathbb{1}$ and $\sigma_y\mathbb{1}\sigma_y$. Furthermore, the encoded spin-flip and phase-shift operators equal $\sigma_x\sigma_x\sigma_x$ and $\sigma_z\sigma_z\sigma_z$, respectively.

*Example 5.* The encoded spin-flip operators for the classical convolutional code $|k_1,k_2,\ldots\rangle\mapsto\otimes_{i=0}^{+\infty}|k_i+k_{i-2},k_i+k_{i-1}+k_{i-2}\rangle$ are given by $\sigma_x\sigma_x\mathbb{1}\sigma_x\sigma_x\sigma_x\mathbb{1}\mathbb{1}\cdots$, $\mathbb{1}\mathbb{1}\sigma_x\sigma_x\mathbb{1}\sigma_x\sigma_x\sigma_x\mathbb{1}\mathbb{1}\cdots$, $\mathbb{1}\mathbb{1}\mathbb{1}\mathbb{1}\sigma_x\sigma_x\mathbb{1}\sigma_x\sigma_x\sigma_x\mathbb{1}\mathbb{1}\cdots$, and so on. The encoded phase shift operators are given by $\mathbb{1}\mathbb{1}\sigma_z\sigma_z\mathbb{1}\mathbb{1}\cdots$, $\mathbb{1}\mathbb{1}\mathbb{1}\mathbb{1}\sigma_z\sigma_z\mathbb{1}\mathbb{1}\cdots$, and so on. Besides, the stabilizer for this code is generated by $\sigma_z\sigma_z\mathbb{1}\mathbb{1}\cdots$, $\mathbb{1}\sigma_z\mathbb{1}\sigma_z\sigma_z\mathbb{1}\mathbb{1}\cdots$, $\sigma_z\sigma_z\mathbb{1}\sigma_z\mathbb{1}\sigma_z\sigma_z\sigma_z\mathbb{1}\mathbb{1}\cdots$, $\mathbb{1}\mathbb{1}\sigma_z\sigma_z\mathbb{1}\sigma_z\mathbb{1}\sigma_z\sigma_z\sigma_z\mathbb{1}\mathbb{1}\cdots$, $\mathbb{1}\mathbb{1}\mathbb{1}\mathbb{1}\sigma_z\sigma_z\mathbb{1}\sigma_z\mathbb{1}\sigma_z\sigma_z\sigma_z\mathbb{1}\mathbb{1}\cdots$, and so on.

*Lemma 3.* Let $C$ be a classical binary code. And let $C'$ be the code obtained by locally Fourier transforming each qubit in the code $C$. Then, both $C$ and $C'$ are stabilizer codes.

*Proof.* Lemma 2 says that $C$ is a stabilizer code. Suppose $g_i$ are the generators of the stabilizer of $C$, and $\overline{\sigma}_{x,j}$ is the encoded spin-flip operator of $C$ as described in lemma 2. Define $g_i'$ to be $g_i$ with $\sigma_x$ replaced by $\sigma_z$. Similarly, I define $\overline{\sigma}_{x,j}'$ to be $\overline{\sigma}_{x,j}$ but with $\sigma_x$ replaced by $\sigma_z$. Then, it is easy to verify that $g_i'$ generate the stabilizer of $C'$. Besides, $\overline{\sigma}_{x,j}'$ is the encoded spin-flip operator for the code $C'$.

*Lemma 4.* Let $C$ and $C'$ be the codes as described in lemmas 2 and 3. Then the code $C''$ obtained by first encoding the state by $C'$ and then encoding the resultant state by $C$ is a stabilizer code. (Compare with Ref. [34] for a similar result.) In addition, each encoded spin-flip and phase-shift operator for $C''$ acts on a finite number of qubits provided that $C$ and hence $C''$ are noncatastrophic.

*Proof.* Suppose $C$ and hence also $C'$ are $k$-input $n$-output codes with finite memory. Then from lemmas 2 and 3, I can write the generators of the stabilizer code $C$ as $A_{i1}A_{i2}A_{i3}\cdots$ where $A_{ij}\in\{\mathbb{1},\sigma_x\}$. Moreover, I write $B_{i1}B_{i2}\cdots$ as the generators of the stabilizer code $C'$ where $B_{ij}\in\{\mathbb{1},\sigma_z\}$. Suppose $X_{i1}X_{i2}\cdots$ and $Z_{i1}Z_{i2}\cdots$ are the encoded spin-flip operators for codes $C$ and $C'$, respectively. Recall that $C$ can be expressed in the form [17]

$$|x_1,x_2,\ldots\rangle\mapsto\overset{+\infty}{\underset{i=1}{\otimes}}\left|\sum_j a_{ij}x_j\right\rangle. \tag{20}$$

Since $C$ is of finite memory, the sum in each of the encode qubits in Eq. (20) above is finite. More precisely, $a_{ij}=0$ or 1 and for each fixed $i$; only a finite number of $a_{ij}$ equals 1. Consequently, the QCC $C''$ can be expressed in the form

$$|x_1,x_2,\ldots\rangle\mapsto\overset{+\infty}{\underset{i=1}{\otimes}}\left[\sum_{p_1,p_2,\ldots}\omega_N^{\sum_j a_{ij}x_jp_i}\left|\sum_r b_{ir}p_r\right\rangle\right], \tag{21}$$

where $b_{ij}=0$ or 1 and for each fixed $i$; only a finite number of $b_{ij}$ equals 1.

If $C$ is catastrophic, its decoding circuit can be expressed as a formal power series matrix. While if $C$ is noncatastrophic, its decoding circuit can be expressed as a polynomial matrix [25]. (See also theorem 1.) Consequently, there exists $c_{ij}\in\{0,1\}$ such that $\sigma_x^{c_{i1}}\sigma_x^{c_{i2}}\sigma_x^{c_{i3}}\cdots$ is an operator acting on the code word of $C''$ whose result is to map $p_i$ to $p_i+1$ for all $i$. Besides, $\sigma_z^{c_{i1}}\sigma_z^{c_{i2}}\sigma_z^{c_{i3}}\cdots$ is an operator acting on the code word of $C''$ whose result is to multiply the code word by a phase $(-1)^{p_i}$ for all $i$. Similarly, there exists $d_{ij}\in\{0,1\}$ such that $\sigma_x^{d_{i1}}\sigma_x^{d_{i2}}\cdots$ is an operator acting on the code word of $C$ whose result is to map $x_i$ to $x_i+1$, and that $\sigma_z^{d_{i1}}\sigma_z^{d_{i2}}\cdots$ is an operator acting on the code word of $C$ whose result is to multiply the code word by a phase $(-1)^{x_i}$ for all $i$. Furthermore, for each fixed $i$, only a finite number of $c_{ij}$ and $d_{ij}$ equals 1 if $C$ is noncatastrophic.

Once I know how to add 1 to $x_i$ and $p_i$ as well as how to add phases $(-1)^{x_i}$ and $(-1)^{p_i}$ to the code words of $C$ and $C'$ in the previous paragraph, I can use them to construct the encoded spin-flip and phase-shift operators for the code $C''$. They are given by

$$\overline{\sigma}_{x,i}''=\sigma_z^{\sum_j d_{ij}c_{j1}}\sigma_z^{\sum_j d_{ij}c_{j2}}\sigma_z^{\sum_j d_{ij}c_{j3}}\cdots \tag{22a}$$

and

$$\overline{\sigma}_{z,i}''=\sigma_x^{\sum_j d_{ij}c_{j1}}\sigma_x^{\sum_j d_{ij}c_{j2}}\sigma_x^{\sum_j d_{ij}c_{j3}}\cdots, \tag{22b}$$

respectively.

After identifying the encoded spin-flip and phase-shift operations in $C''$, it remains for me to find the generators of the stabilizer of $C''$. First, by direct checking, I know that the operator $\sigma_x^{c_{i1}}\sigma_x^{c_{i2}}\cdots$ belongs to the stabilizer of $C''$ for all $i$. Then, similar to the proof of lemma 2, I consider operators in the form $A_{i1}A_{i2}\cdots$ with $A_{ij}\in\{\mathbb{1},\sigma_z\}$ that commute with the encoded spin flip, encoded phase shift, and $\sigma_x^{c_{i1}}\sigma_x^{c_{i2}}\cdots$. Now, I choose a (countable number of) generators among them. Then, the union of these operators and $\sigma_x^{c_{i1}}\sigma_x^{c_{i2}}\cdots$ generates a stabilizer of the code $C''$. ■

*Example 6.* When $N=2$, the encoded spin-flip operators for the QCC in Eq. (1) are

$$\sigma_z\sigma_z\sigma_z\sigma_z\sigma_x\sigma_x\mathbb{1}\mathbb{1}\sigma_z\sigma_z\sigma_z\sigma_z\mathbb{1}\mathbb{1}\cdots,$$

$$\mathbb{1}\mathbb{1}\mathbb{1}\mathbb{1}\sigma_z\sigma_z\sigma_z\sigma_z\sigma_x\sigma_x\mathbb{1}\mathbb{1}\sigma_z\sigma_z\sigma_z\sigma_z\mathbb{1}\mathbb{1}\cdots,$$

and so on. In addition, the encoded phase-shift operators are

$$\sigma_x\sigma_x\sigma_x\mathbb{1}\sigma_x\mathbb{1}\sigma_x\sigma_x\mathbb{1}\mathbb{1}\cdots,$$

$$\mathbb{1}\mathbb{1}\mathbb{1}\mathbb{1}\sigma_x\sigma_x\sigma_x\mathbb{1}\sigma_x\mathbb{1}\sigma_x\sigma_x\mathbb{1}\mathbb{1}\cdots,$$

and so on.

According to the proof of lemma 4, fault-tolerant computation is possible for all QCCs constructed using the method in theorem 2. More importantly, if one starts with a noncatastrophic classical convolutional code $C$, then the fault-tolerant spin-flip, phase-shift, and controlled swapping for the QCC $C''$ constructed in theorem 2 can all be done in a finite number of quantum gates. In fact, as long as I carefully wire my quantum circuit to prevent the spreading of quantum errors throughout all the qubits (see Ref. [30] for tips on how to do this), I can perform fault-tolerant quantum computations on this kind of QCCs. Suppose I have a quantum signal $|x_1,x_2,\ldots\rangle$, and if I follow the fault-tolerant computation wiring rule, I may even perform computations between the $i$th and $j$th encoded qubits in the above signal provided that their encoded spin-flip and phase-shift operators acts on distinct places in the encoded signal.

## V. DISCUSSION

In summary, I have generalized the VDA to the QCC and have shown the optimality of the QVA for a memoryless channel. In addition, I reported a simple way to test if a QCC is noncatastrophic. The key observation for all these is that a lot of classical coding concepts can be ''quantized'' provided that one performs the relevant operations with care. Finally, I show that certain QCCs can perform fault-tolerant quantum computations. Since classical convolutional codes may be regarded as stabilizer codes and good classical convolutional codes exist, therefore I conclude that good QCCs and their decoding algorithm exist.

## ACKNOWLEDGMENTS

I would like to thank Debbie Leung for her useful comments. This work was supported by the Hong Kong Government under RGC Grant No. HKU 7095/97P.

[1] P. W. Shor, Phys. Rev. A **52**, R2493 (1995).
[2] R. Laflamme, C. Miquel, J. P. Paz, and W. H. Zurek, Phys. Rev. Lett. **77**, 198 (1996).
[3] A. M. Steane, Phys. Rev. A **54**, 4741 (1996).
[4] A. R. Calderbank, E. M. Rains, P. W. Shor, and N. J. A. Sloane, IEEE Trans. Inf. Theory **44**, 1369 (1998).
[5] A. R. Calderbank, E. M. Rains, P. W. Shor, and N. J. A. Sloane, Phys. Rev. Lett. **78**, 405 (1997).
[6] D. Gottesman, Ph.D. thesis, Caltech, 1997 (unpublished).
[7] H. F. Chau, Phys. Rev. A **55**, R839 (1997).
[8] H. F. Chau, Phys. Rev. A **56**, R1 (1997).
[9] E. M. Rains, R. H. Hardin, P. W. Shor, and N. J. A. Sloane, Phys. Rev. Lett. **79**, 953 (1997).
[10] S. L. Braunstein, Phys. Rev. Lett. **80**, 4084 (1998).
[11] S. Lloyd and J.-J. E. Slotine, Phys. Rev. Lett. **80**, 4088 (1998).
[12] E. Knill, e-print quant-ph/9608048.
[13] H. F. Chau, Phys. Rev. A **58**, 905 (1998).
[14] A. J. Viterbi, IEEE Trans. Inf. Theory **13**, 260 (1967).
[15] A. J. Viterbi, IEEE Trans. Commun. Technol. **19**, 751 (1971).
[16] G. D. Forney, Jr., IEEE Trans. Inf. Theory **16**, 720 (1970).
[17] A. Dholakia, *Introduction to Convolutional Codes with Applications* (Kluwer, Boston, 1994), Chap. 5.
[18] J. K. Omura, IEEE Trans. Inf. Theory **15**, 177 (1969).
[19] G. D. Forney, Jr., Inf. Control. **25**, 222 (1974).
[20] A. J. Viterbi, IEEE Trans. Inf. Theory **13**, 260 (1967).
[21] R. M. Fano, IEEE Trans. Inf. Theory **9**, 64 (1963).
[22] H.-K. Lo and H. F. Chau, Science **283**, 2050 (1999) and the associated supplementary materials available at URL: http://www.sciencemag.org/feature/data/984035.shl.
[23] D. Gottesman (unpublished); see also e-print quant-ph/9802007.
[24] M. Reck, A. Zeilinger, H. J. Bernstein, and P. Bertani, Phys. Rev. Lett. **73**, 58 (1994).
[25] J. L. Massey and M. K. Sain, IEEE Trans. Comput. **17**, 330 (1968).
[26] J. L. Massey and M. K. Sain, IEEE Trans. Autom. Control. **12**, 644 (1967).
[27] A. Yu. Kitaev, Russ. Math. Surv. **52**, 1191 (1997).
[28] D. Aharonov and M. Ben-Or, *Proceedings of the 29th Annual ACM Symposium on the Theory of Computing* (ACM, New York, 1998), p. 176.
[29] E. Knill, R. Laflamme, and W. Zurek, Science **279**, 342 (1998).
[30] J. Preskill, in *Introduction to Quantum Computation and Information*, edited by H.-K. Lo, T. Spiller, and S. Popescu (World Scientific, Singapore, 1998), p. 213.
[31] D. Gottesman, Phys. Rev. A **54**, 1862 (1996).
[32] D. Gottesman, Phys. Rev. A **57**, 127 (1998).
[33] A. M. Steane, Philos. Trans. R. Soc. London, Ser. A **356**, 1739 (1998).
[34] D. Gottesman, e-print quant-ph/9607027.